COS 221 Fundamental Data Structures Homework #5

Dimitar Kumanov ID# 100101031 e-mail: [DAK140@aubg.edu](mailto:DAK140@aubg.edu)

Explanation of the results:

A single run of the program produced the following results:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | BST | AVL | B-tree; M=7 | |
| Within nodes | Jumps |
| 1000 | 17508 | 14625 | 16129 | 5100 |
| 10 000 | 235294 | 196829 | 215549 | 66294 |
| 20 000 | 524675 | 423930 | 460752 | 139340 |

The results from the table show that the number of comparisons/jumps for AVL trees are equal to about 80% the number of comparisons/jumps for a standard Binary Search Tree (with the two smaller vectors the percentage is actually 83%). The number of comparisons within nodes for the B-tree are roughly equal to 90% the number of comparisons/jumps used in the BST (for the two smaller vectors the percentage is 92% and for the biggest vector it is 87%). On the other hand, the number of jumps made in the B-tree is only about 27% the number of jumps made in the BST, which proves the theory that B-trees are better suited for working with nodes that store large blocks of data. In these cases, accessing the data would be an expensive and slow operation and minimizing the number of access times (jumps) would greatly increase the working efficiency of making manipulations in the tree (insertion, deletion, searching, etc.).